DOCUMENT RESUME

ED 261 644 IR 011 790

AUTHOR McClain, Gary R.
TITLE Lost in the Forest: Overcoming the Need to Pontificate in Course Development.
PUB DATE 4 Jun 85
NOTE 14p.; Paper presented at the Computer-Based Training Conference (Dallas, TX, March 17-20, 1985).
PUB TYPE Guides - Non-Classroom Use (055) -- Viewpoints (120) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Science Education; Course Descriptions; *Data Processing; Educational Strategies; Guidelines; *Instructional Design; *Sequential Approach; Skill Development; *Task Analysis

ABSTRACT

Designed to illustrate effective course design for user-oriented materials for data processing education or training, this paper argues that conceptual knowledge should be integrated into a course only as it is needed to support task knowledge. The importance of determining the direction of a course by planning and controlling the informational content is emphasized, and an end-user fourth-generation language (4GL) course is described to illustrate the differences between conceptual and task knowledge and how they should be integrated. Step-by-step guidelines for sequencing information in a course unit to achieve this type of integration are given: (1) isolate the tasks; (2) isolate the concepts; (3) put the tasks in order; (4) integrate the information; (5) number the objectives; and (6) create an outline. The conclusion suggests that since there is never a prototypical consumer, painstaking initial design is the only way of avoiding information overload in a training package or educational program, and any information which is superfluous should be ignored. (JB)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)
X This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED261644

Lost in the Forest:
Overcoming the Need to Pontificate in Course Development

Gary R. McClain, Ph.D.

Infodata Systems Inc.

The over-used cliche, regarding the inability to "see the forest for the trees," can be all too often applied when describing the content of courses for user education. Students are force-fed everything they "ever wanted to know" with the true purpose of the training -- the acquisition of usable skills -- included as an afterthought.

This problem is apparent throughout the educational media. Here are a couple of examples.

A computer-based training (CBT) course designed to teach the basic use of one of the mainframe operating systems begins with a lengthy conceptual discussion of the computer, how it is designed, the operating environment, etc. Numerous abbreviations are included. By the time the student finally progresses to the material concerning the actual "how-tos" of this system, the initial information, particularly the abbreviations, has been forgotten.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Gary R. McClain

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR011790

Students attending a three-day classroom course to learn how to use a software product for an accounting application spent the first day listening to an in-depth overview of the design and interfaces of the system. This included database and other conceptual information. Not having been exposed to the use of the product, students felt overwhelmed by the end of the day.

These two situations serve to illustrate a crucial oversight in user-oriented data processing education. The word use is ignored and students are buried under a veritable plethora of concepts, explanations, and definitions. Learners are "blown away" by -- albeit well-meaning -- pontification. If the teacher is unable to see the forest for the trees, then how is the student supposed to?

The cosmetics seem to be intact, including good screen design for a CBT course or informative overheads for a classroom course, appropriately written objectives and absolutely "tons" of content. That missing piece in what should be a good course occurred in a crucial point in the course development process: the initial design.

All too often, courses are the result of "ploughing" into the materials, covering every last detail, and then squeezing the results into the allotted time frame. It's no wonder that everyone loses.

We often develop courses with the same approach that we were exposed to in our own educational processes -- heavy "everything you ever wanted to know" up front conceptual detail, with the "how-tos" thrown in as an afterthought. It's like going through twelve years of public education, four years of liberal arts college, followed by a month or two of on-the-job orientation. Many of us went through that type of training, so we're turning around and doing the same thing to our users.

The direction of a course is, hopefully, determined and adhered to during the initial design. So it is at this important stage that the informational content must be both planned and controlled.

A key to this control is the integration of conceptual knowledge, such as definition, from task knowledge, the how-tos or skills. The types of information are separate, yet their integration is crucial to the development of a well-balanced course.

Concept Support Tasks

We often get so caught up in our lecture about why something "is the way that it is" that the real reason our students are in class in the first place -- to learn how to complete real-life job tasks -- is neglected.

But the truth is that conceptual information is not an end in and of itself -- it hasn't been since graduation from the college of liberal arts. Students come to a data processing education class to learn how to use the computer to do their jobs a bit easier. And this must be the "raison d'etre" of the course. Any information other than how to use that system is strictly peripheral.

This can be illustrated by using the teaching of an end-user fourth-generation language (4GL) course as an example. The content of a course such as this can easily be divided into concepts and tasks.

As stated previously, conceptual knowledge includes definitions and underlying facts. In a fourth-generation language course, this knowledge would include not only a product overview, but definitions of words such as database, report and sequential access. It would also include background information such as file structure, or design considerations.

Task knowledge, or how-tos, encompasses the actual use of the product. With a 4GL course, this would include opening the database, performing a sequential search, and formatting a report.

Clearly, users are most likely concerned more with the tasks at hand rather than with the conceptual knowledge, since their job performance literally depends on successful execution of these tasks. Yet, the concepts necessary to the performance of these skills are also crucial.

Courses are certainly organized with both types of knowledge in mind. But the answer is not to separate them into camps, with strict attention to the concepts followed by a presentation of the tasks. The separation is necessary, but should instead lead to an integration that focused on the true purpose of a training course for users: use of the system.

The best rule of thumb to guide the initial course design is that conceptual information is only needed if it supports a task. Furthermore, this type of information, if deemed useful, is only needed when supporting a specific task.

In other words, isolate the two types of information and then integrate them into the course with tasks being the focus.

Referring back to the 4GL example, two approaches to the sequence of information can be compared. Assume that end-users are being taught the basic database concepts as well as sequential access.

The information could be simply sequenced first with concepts, followed by tasks:

Concepts

Definition of a database

Definition of terms

Record
Field
Subfield
Sequential search
Report

Tasks

Basic 4GL syntax -- sequential search

Formatting a report

An integration of the conceptual information with the actual tasks might be sequenced quite differently (the concepts are indented under the tasks which they support).

Demonstration of a sequential search database.

Sequential search

Report

Basic 4GL syntax --sequential search

Record

Formatting a report

Field

Subfield

From an "old school" approach, this integration seems backwards. But look at it this way -- we tend not to remember information until we need it. Why barrage students with terms such as record and database until they have become familiar with the tasks -- the actions that must be performed? Sequential searching, for example, is an "academic" concept until one has been exposed to both the process and results of a sequential search. Furthermore, the term "record" is much more meaningful when discussed in the context of attempting to get useful information from one. And why not discuss "fields" in light of how the data from a field is displayed on a report? The facts are much more relevant when inserted at key points.

Notice that the second example of the sequence of information begins with a demonstration of a sequential search. Don't submit learners to a "big-buildup" and allow them to form misconceptions which must later be dispelled. A course is similar to a Christmas tree, and the initial example provides students with an image, or an end goal. With this image in mind, learning is merely a matter of "fleshing in" the framework.

This initial example might consist of a simple 4GL command followed by a report. The example is then used repeatedly until any gaps in understanding have been filled in with "whys" and "hows". The focus of the learning is thus the tasks which form the "building blocks" of the course. Concepts and definitions are the mortar which makes the building blocks hold together.

Separate, then Integrate

The sequencing of tasks with supportive concepts requires extensive initial planning. This planning should result in a strong detailed outline. Without this outline, the risk of "ploughing into the forest without a map" results, with the inevitable false starts and changes of direction.

The standard approach to course development, with the global

objective and course units consisting of individual objectives, does not need to be radically changed. What may need some "fine tuning" are the focus of these objectives and, most likely, the sequence of both objectives and units.

Again referring to the 4GL example, suppose that a basic searching and reporting course is being developed, with units such as "Sequential Searching", "Computing", and "Report Generation". One at a time, each unit needs to be broken down into the components -- the tasks -- which are the objectives of the unit. The concepts are considered after this process.

The following are some step-by-step guidelines for sequencing the information in a course unit.

1. Isolate the Tasks

   This can be accomplished by looking at what tasks need to be taught in the unit, and listing them. It may help to write each task on a notecard, so that the cards can be shuffled around as the tasks are put into some kind of order. Tasks in a "sequential searching" unit might include, for example, "Opening the database", "Searching without search criteria", and "Searching with search criteria".

2. Isolate the Concepts

Keeping in mind only the unit topic, brainstorm for the concepts, explanations, definitions, etc., which relate to the unit topic. This is a separate process from isolating the tasks, because they will be "meshed" in the next step. For the "Sequential Searching" unit, this would include an explanation of this type of search with definitions of words such as field and record.

3. Put the Tasks in Order

Now that the informational content of the unit has been ascertained, determine a logical order from the tasks. This is generally chronological, but may be organized simply by topic if chronology is not required.

4. Integrate the Information

At this point the content can be "fleshed in" by deciding where to insert conceptual knowledge. This information should be inserted only when it supports the ensuing objective. It may be helpful to spread the notecards out on a large surface, and arrange them as a flowchart.

5. Number the Objectives

The objectives of the unit should consist of whatever tasks are included. This results in objectives which are truly action-oriented. If, for example, five tasks have been isolated, then your unit should consist of five objectives.

6. Create an Outline

Turn the makeshift flowchart into a formal outline, with the objectives as headings and the concepts listed underneath them. This is a crucial guide for the actual course development which follows and will help to avoid the informational overload tasks.

Less is More

There is extensive research with accompanying statistical evidence which supports the theory that much of what we are told, particularly in a lecture situation, goes "in one ear and out the other". Regardless of the validity of this evidence, most educators, not to mention learners, can attest to this phenomenon based merely on common sense. Human beings screen out information which is not relevant at the moment, no matter how valuable this information may seem to the instructor.

This places a special burden on the educator in a data processing situation. Whether teaching specific applications or more generic skills, such as a fourth-generation language, the instructor is faced with a variety of considerations. These include students with backgrounds ranging from "never having touched a keyboard", to those with extensive data processing backgrounds. A further complication is the many diverse purposes that users may have for the same product. Software and hardware vendors generally find that each installation of their product has its own set of unique characteristics.

The reality of the learning situation, whether a classroom or a CBT course, is that there is almost never a prototype consumer. One training package must meet the needs of broad ranges of background and purposes. Thus, it is no surprise that educators often find themselves getting lost in the forest. We simply cannot design courses which are everything to every possible learner! Information overload is the inevitable result.

Painstaking initial design is a way of avoiding this pitfall. Students enter into a learning situation for a variety of reasons, but the all encompassing objective can be summarized very simply: to learn one or more aspects of the use of a product. Any information which does not directly contribute to an understanding of the use of this product is superfluous and should be ignored.

When we overload students with concepts -- system design, inner-workings, data processing theory, accounting practice, and whatever else strokes the instructor's ego, we risk the frustration and boredom which result.

Conceptual information is certainly the basis of "real world" application. Combining an appropriate amount of the conceptual information with the overriding purpose of a training course -- to learn to use a product -- can result in an exciting synergy. The learner is familiarized with a task and his or her curiosity is piqued, enough conceptual information is provided to support the task, and then the learner is guided a step further. With this emphasis on specific tasks, the learner is allowed enough space and flexibility to subsequently apply the knowledge to his or her own needs and experiences. These needs and experiences are the basis for learning, not the end result.

As data processing educators we are in the important position of literally 'ushering in' the technological revolution. For many we provide the first exposure to the computer, and unlock its mysterious wonders. It is our responsibility to foster comprehension rather than awe in our learners. We must facilitate the acquisition of knowledge that can be generalized to real world usage, as well as a desire to learn even more.